Adaptive Reliable Routing Protocol for Wireless Sensor Networks

Nourhene Maalel, Pierre Roux, Mounir Kellil
CEA, LIST, Communicating Systems Laboratory
Gif-sur-Yvette, France
{nourhene.maalel,pierre.roux,mounir.kellil}@cea.fr

Abdelmadjid Bouabdallah
UTC, Heudiasyc Laboratory, UMR CNRS 6599,
Compiègne, France
madjid.bouabdallah@hds.utc.fr

***Abstract*—Many Wireless Sensor Networks (WSN) applications success is contingent upon the reliable delivery of high-priority events from many scattered sensors to one or more sink nodes. In particular, WSN has to be self-adaptive and resilient to errors by providing efficient mechanisms for information distribution especially in the multi-hop scenario. To meet the stringent requirement of reliably transmitting data, we propose a lightweight and energy-efficient joint mechanism for packet loss recovery and route quality awareness in WSNs. In this protocol, we use the overhearing feature characterizing the wireless channels as an implicit acknowledgment (ACK) mechanism. In addition, the protocol allows for an adaptive selection of the routing path, based on a collective cooperation within neighborhood.**

Keywords- WSNs; Reliable transport; Routing

I. INTRODUCTION

All WSN (Wireless Sensor Network) technologies have experienced an exponential increase in popularity mostly due to their potentially low cost of maintenance and deployment.

However, wireless sensor networks may face a number of challenges that can hamper their widespread exploitation [1]. A WSN has to be self-adaptive and resilient to errors by providing efficient mechanisms for information distribution especially in the multi-hop scenario. These requirements have to be achieved in a networking environment that is constrained by limited processing capability, scarce energy resources and unreliable communication channels [1]. In particular, in a typical harsh environment, the radio signal is often affected by interference: medium access conflicts, multipath fading, shadowing, etc. These problems may result in significant packet losses in WSNs. Moreover, the success of many applications (particularly mission-critical ones like life-care data and alarms) requires the delivery of high-priority events to sinks without any loss from the original sources to the final destination [2]. These constraints emphasize the need for an energy-efficient, scalable and reliable data transport system.

Data retransmission has been considered as one of the most common schemes [3] for improving transmission reliability in WSN. ACKnowledgment/ Negative ACKnowledgment (ACK/NACK) messages are the basic method used to assess the necessity of retransmission. Nevertheless, such a method generates an extra traffic causing an additional overhead, which is not suitable in a highly constrained and error prone environments, like WSNs. Accordingly, an alternative solution should be found to deal with retransmissions without wasting bandwidth.

In this paper, we define a reliable and energy-efficient joint mechanism, for packet loss recovery and route quality evaluation in WSNs. In this protocol, we use the overhearing feature, characterizing the wireless channels [3], as an implicit ACK mechanism. In addition, the protocol allows for an adaptive selection of the routing path based on a link state metric.

The remainder of this paper is organized as follows: the next section highlights the need for reliable data delivery in WSNs, and reviews solutions aiming at providing it. Protocol description and analysis are given in Section 3, and finally, Section 4 concludes this paper.

II. BACKGROUND

The error control can be implemented as multipath routing by forwarding packets along several paths in order to improve the overall reliability [4]. Copies of the same packets can be forwarded randomly over multiple routes [4]. Another solution is to identify many paths and select one as the primary route while the other alternatives are used in case of problems in the primary path [5]. Maintaining multiple paths is usually costly in large scale WSNs.

Another traditional way to achieve reliable transmission is the Automatic Repeat reQuest (ARQ) mechanism based on the ACK/NACK messages [6]. However, this mechanism should be minimized because sensor nodes are severely resource constrained and data transmission is one of the most costly operations performed by sensors [6]. Moreover, the unreliable radio channel affects the acknowledgment delivery as well. If the sender does not receive any acknowledgment in the specified time interval, it retransmits the message even if the packet was properly delivered. In practice, the sender node makes a delimited number of trials to successfully deliver a message. Therefore, relying on explicit acknowledgement is not appropriate with regard to the constrained nature of WSNs.

More recently, a Multicast Protocol for Low power and Lossy Networks (MPL) called Trickle Multicast [7], was designed. Trickle multicast utilizes a sequence number in the data packet to cope with packet losses. Packets along with their respective sequence numbers are temporarily stored by

 ISBN: 978-1-61208-284-4

the nodes so as retransmission can be triggered when necessary. Trickle multicast, though is based on network flooding for data dissemination and storage. Given the resource constraint nature of WSNs, this flooding mechanism is not suitable to sensor networks.

We could identify two categories of transmissions Hop By Hop (HBH) and the End to End approach (ETE). According to She and al. [9], HBH is more energy efficient at the cost of large transmission delay compared to ETE. Nevertheless, HBH outperforms ETE on the delay metric for high bit error rate cases. Given that Zhao et al. [8] show that error rates of 10% or above in dense WSN may be experienced, HBH is the most suitable candidate for WSNs. Let's notice that the problem with ETE recovery is highly related to the harsh radio environments of deployment and to the multi-hop forwarding techniques, which favor exponential error accumulation [8].

Some researches proposed solutions to alleviate the retransmissions cost like PSFQ [10], which distribute data from a source node by sending data at a relatively slow speed but allowing nodes that come across data loss to recover any missing segments from their local immediate neighbors. This protocol is efficient for fast recovery but if packet lost occurs in an intermediate node towards the sink, buffer must standby until packet re-transmission is done. This causes buffer overflow and increases data transmission delays. Blagojevi and al. [11] presented a probabilistic acknowledgement mechanism switching between explicit and implicit acknowledgement depending on the current path reliability. For this solution, path reliability is determined by measuring the Received Signal Strength Indication (RSSI) which is proved to not always be a good indicator to estimate the link state [12]. Messina and al. [13] proposed a solution where the protocol achieves reliability through cashing and retransmission. As mentioned, this solution requires each one hop neighbor to cash the data until the success of its transmission. Once a packet loss is detected, all the one hop neighbors will act on the behalf of the node which experiences the loss by retransmitting the packet and performing its routing task. Such a practice leads to extra energy consumption and may fasten nodes "battery depletion".

III. PROPOSED PROTOCOL

A. Overview of the mechanism

Our solution seeks into elaborating an efficient error control mechanism with implicit acknowledgments to face the link failure and packet loss problem in WSN. When a sensor node transmits a packet, nodes of its neighborhood overhear its packet transmission even if those are not the intended recipients [3]. This arises from the broadcast nature of the wireless channel.

Our solution uses this overhearing characteristic instead of the acknowledgment messages to guarantee reliability on networks. Moreover, when a packet loss is detected, retransmission is carried out by the most reliable link between the node which sent the (lost) packet and its one-hop neighbors. The reliability of links is defined according to a metric which will be detailed in the next section. Our algorithm relies on a spanning tree for ordinary routing operations, and resorts to exploiting alternative paths only when a malfunctioning is detected.

B. Protocol operation

1) Considered architecture

We consider a dense and randomly distributed WSN. Before discussing the details of the protocol, we need to clarify our assumptions:

-All nodes have sufficient resources to carry their sensing, computing, and transmission/reception operations.

-Data packet is generated by sensors and transmitted to the sink node.

- Each sensor node is stationary for its lifetime and is able to record the link performance between itself and its neighboring node in terms of number of lost packet / number of sent packets.

-We adopt a routing scheme in which the routing decision takes the shortest path towards the sink. Each node is assigned a rank corresponding to the hop-distance to the sink and data is carried rank by rank towards the sink. In this sense, the node B in Figure 1 has rank N and its neighbors have rank N-1, N, or N+1. We assume that each node is aware of its own rank (in respect to its neighbors) as well as the ranks of its neighbors.

-Each node of rank N (Figure 1) classifies its k neighbors of rank N-1 from index 0 to k with 0 corresponding to the most reliable node according to our metric defined in the next section. The node with index 0 is the elected one to carry out the retransmission task when packet loss is detected.

2) Index assignment

Our protocol relies on its routing metric to assign indexes to nodes. As mentioned above, index 0 corresponds to the most reliable link (the higher metric). The index assignment is used to choose the best next hop for the packet retransmission hence its importance. The metric component of our protocol evaluates links according to the Link Quality Indicator (LQI) and the probabilistic history model. The LQI is a metric of the current received signal quality. This measurement is reported with each received packet in the MAC header by the used 802.15.4 standard [14]. The use of LQI ensures adaptability to the environmental conditions by expressing the real quality of the link. Besides, LQI experiences frequent fluctuations in highly interfered environment. Hence, we consider statistics (average number of lost packet per link) as a basis to assess the reliability of links.

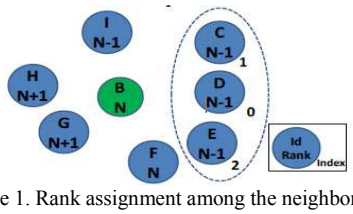

Figure 1. Rank assignment among the neighborhood

For this reason, we have decided to weight the metric by the link failure probability given by our probabilistic history model, Phist BC. Since the channel state is binary (packet received: Up, packet lost: Down), a simple count of the number of state is sufficient to fully describe the history:

$$P_{hist\ BC} = \frac{nUp}{nUp + nDown} \quad (1)$$

Therefore, even if the last LQI value recorded does not match the real state of the link we can correct it. Let's precise that *Phist BC* is set to 1 in the beginning (during a fixed time Tinit) before getting effective history. In fact, in the establishment of the process, we do not have sufficient feedback to assess the reliability of a link. To do so, our protocol assigns each link a cost given by the following expression:

$$Metric\ \ (B,C) = K * e^{-age} * LQI_{BC} + \frac{1}{P_{hist\ BC}} \quad (2)$$

where LQI_{BC} denotes the link state indicator between nodes B and C and *age* corresponds to the delay since the LQI value has been recorded. The exponential function provides a decreasing function according to the age, which means that more recent values of LQI are considered as more significant. The $P_{hist\ BC}$ represents the probability of link success between nodes B and C. *K* is a constant used to weight the equation. These metrics are calculated periodically in the network to update the index assignments and make the protocol more robust to the environmental change. This update period depends on the packet error rate of the network: the faultier the network is, the more frequent the update occurs.

3) Algorithm description

Figure 1 represents a node B of rank N and its neighborhood. More particularly, it shows its N-1 neighbors C, D and E of index 1, 0, and 2, respectively. Let's notice that our protocol provides uniqueness of index to avoid collision problem: When different nodes have the same index, a random back off is added to the metric in order to have distinguish index. Once a packet with a Packet IDentifier (PID) is received for the first time by a node of rank N from a node N+1, a transient context is created in its memory to manage this packet PID.

This context includes packet content and PID in order to allow possible retransmissions. If the node has index 0 relatively to the sender node, the context is considered as a 'Primary' one (P Ctxt) and the packet is immediately forwarded. Otherwise, the context is considered as 'Secondary' (S Ctxt) and the packet is cached waiting for a possible retransmission request.

To make it clearer, we will consider 3 scenarios shown in Figure 2. In the loss-free case (Figure2/a), all the nodes C, D and E receive the packet. Node D (which has index 0 for B), creates a primary context for PID, while node C and E (which have an index greater than 0 for B) create a secondary context for packet PID. Because node D has created a primary context for PID, it immediately forwards the packets towards its own neighbors. At this time, the node B also receives the packet forwarded by node D. There is an implicit acknowledgement for packet PID so the node B can release its primary context for PID. After a while, nodes C and E realize that node B didn't send any Explicit Retransmission Request (ERR) message. They can safely get rid of their secondary context for packet PID. In the loss-free case, this process goes on until the packet PID reaches the sink, without involving any waiting period in any of the forwarding nodes on the path to the sink.

Now, if we consider a case including packet loss (Figure 2/b), we can come back to the situation where node B has just received packet PID and has just created a primary context for this packet. Again, it forwards the packet to its neighbors, but we now assume that the node D doesn't receive the packet, while nodes C and E receive it. The node C has index 1 with respect to node B so it creates a secondary context for packet PID. Then the node C waits for a possible Explicit Retransmission Request (ERR) from node B with respect to packet PID. Note that this message is short, as it does not contain the data payload of packet PID.

When the node C receives the ERR message for packet PID, it immediately forwards this packet toward its neighbors, because its index for node B is 1. Once node B receives the implicit acknowledge from node C, it broadcasts an Explicit Retransmission Cancel message (ERC) with respect to packet PID. ERC is a short message similar to ERR. Once this message is received, the node E deletes this message and releases its secondary context for PID. We may now consider another case including packet loss (Figure 2/c). We come back to the same situation as before, but we now assume that among neighbors of rank N-1, only node E has received packet PID from node B. Once Node B detects the packet PID was not forwarded by D, it sends an ERR for PID as before to C, and E. However, node E does not immediately forwards packet PID (even if it is the only node which is able to retransmit the packet) because it is aware that its index for B is 2. That is why it waits for a delay TDelay. Then, if no ERC message with respect to PID has been received from node B, it turns its secondary context for PID into a primary context, and it forwards packet PID to its own neighbors. The rule is that once a node having index *n* with respect to another node receives an ERR from this other node, it waits for a delay equal to *(n-1)* times TDelay for a possible ERC. If no ERC is received during this time, then retransmission occurs.
This process aims to avoid sending duplicate packets and consequently to reduce bandwidth consumption.

C. Protocol Analysis

It should be noted that in case of failure, this algorithm does not solicit the neighbor for which a failure was observed. Our protocol is particularly adapted to packet losses caused by a change in the channel state such as slow

 ISBN: 978-1-61208-284-4

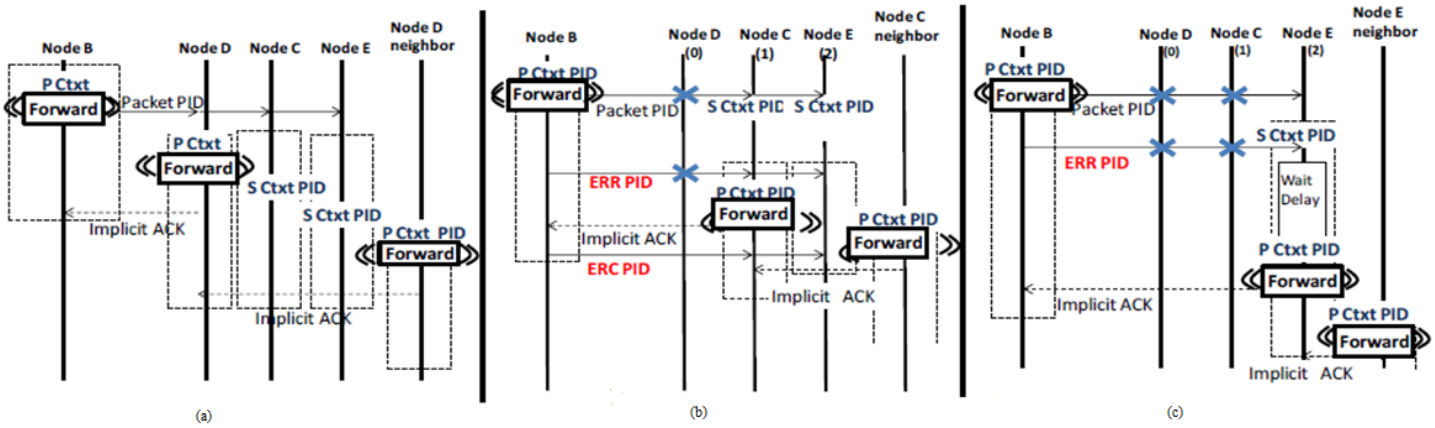

Figure 2. Protocol steps for data forwarding from Node B: (a)Loss free case, (b)Packet loss in Node D case, (c)Packet loss in Node D and Node C case

shadowing that affects the radio link. A retransmission occurring on the same radio link just after a failure is likely to bring a failure again while with the current proposed algorithm, another radio link is exploited. In the case of the set of neighbors of rank N-1 is limited to only one node (sparse network case), this principle doesn't apply, and repetitions should be carried out by the same neighbor.

Besides, packet losses on the implicit acknowledgement messages have not been considered in this paper. With the proposed algorithm, such losses may result in duplicate instances of a packet PID being forwarded up to the sink according to different paths. Given that paths are not disjoint, a node may receive the same packet twice. To alleviate this issue, it is recommended that nodes maintain a list of recently received packet PIDs and to drop packets once a duplicate is detected.

Moreover, the transmission of data from a node to its neighbor must be completed within a specified time. If the packet does not reach the next hop within this time limit, it is dropped and considered as it has been lost.

IV. CONCLUSION AND FUTURE WORK

In this paper, we proposed a lightweight protocol to tackle packet losses in WSNs. We provided a solution based on the implicit ACK mechanism and on an adaptive selection of the routing path based on the link quality evaluation. For future work, we intend to examine additional parameters that influence on the time-varying link reliability, and also plan to evaluate our protocol in comparison to other solutions of the state of the art using simulations.

REFERENCES

[1] Y. Xiao, X. Li, Y. Li, and S. Chen, "Evaluate reliability of wireless sensor networks with OBDD," In Proceeding of IEEE International Conference on Communications (ICC), Dresden, June 2009, pp. 1–5, doi:10.1109/ICC.2009.5199006.

[2] S. Qaisar and H. Radha, "Multipath distributed data reliability for wireless sensor networks," In Proceeding of IEEE International Conference on Communications (ICC), Dresden, June 2009, pp. 330–334.

[3] G. N. Lee and E. N. Huh, "Reliable data transfer using overhearing for implicit ack," ICROS-SICE, Fukuoka 2009, pp. 1976 -1979

[4] B. Deb, S. Bhatnagar, and B. Nath, "ReInForM: Reliable information forwarding using multiple paths in sensor networks," Annual IEEE International Conference on Local Computer Networks. (LCN) , Bonn, October 2003, vol. 0, pp. 406-415.

[5] D. Ganesan, R. Govindan, and S. Shenker, "Highly resilient, energy-efficient multipath routing in wireless sensor networks," ACM SIGMOBILE, October 2001, vol. 5, no. 4, pp. 11–25, doi:10.1145/509506.509514.

[6] I. Stojmenovic and A. Nayak, "Design guidelines for routing protocols in ad hoc and sensor networks with a realistic physical layer," IEEE Communications Magazine, 2005, pp. 101-106.

[7] http://tools.ietf.org/pdf/draft-ietf-roll-trickle-mcast-04.pdf, retrieved July 2013.

[8] J. Zhao, R. Govindan, and D. Estrin, "Computing Aggregates for Monitoring Wireless Sensor Networks," Proceedings of the First IEEE international Workshop on Sensor Network Protocols and Applications, Bonn 2003, pp. 139 – 148.

[9] H. She, Z. Lu, and A. Jantch, « Analytical evaluation of retransmission schemes in wireless sensor network,» In Proceeding of IEEE Vehicular Technology Conference (VTC) Barcelona, April 2009, pp. 1-5.

[10] Campbell, Andrewt T.; Krishnamurthy, and Lakshman, " Pump-slowly, fetch-quickly (PSFQ): a reliable transport protocol for sensor networks," .IEEE Journal on selected area in communications, 2005, pp. 862 – 872.

[11] Miloš Blagojevi, Majid Nabi1, and Marc Geilen1, "A Probabilistic Acknowledgment Mechanism for Wireless Sensor Networks ," IEEE international conference on networking, architecture and storage (NAS), Dalian, July 2011, pp. 63-72.

[12] Shinuk Woo and Hwangnam Kim, "Estimating Link Reliability in Wireless Networks," INFOCOM ,San Diego, March 2010, pp. 1-5.

[13] Messina and Daniele, "Achieving Robustness through Caching and Retransmissions in IEEE 802.15.4-based WSNs," Proceedings of IEEE International Conference on Computer Communications and Networks (ICCCN), 2007 Honolulu, pp. 1117 – 1122.

[14] ZigBee Alliance, http://www.zigbee.org, retrieved July 2013.

 ISBN: 978-1-61208-284-4